

Signature Router & Management System - Product Requirements Document

Author: BMAD Product Manager Agent

Date: 2025-11-26

Version: 1.0

Status: ✓ APPROVED FOR IMPLEMENTATION

Executive Summary

El **Sistema de Enrutamiento y Gestión de Firmas Digitales** es una plataforma bancaria crítica que orquesta múltiples canales de autenticación (SMS, Push, Voice, Biometric) para optimizar costos, experiencia de usuario y resiliencia operacional.

El sistema toma decisiones inteligentes de enrutamiento en tiempo real basadas en reglas de negocio configurables, gestiona fallbacks automáticos cuando un canal falla, y proporciona trazabilidad completa para auditoría y compliance bancario.

Alcance: Sistema backend + API REST + Admin Portal web para gestión de reglas

What Makes This Special

Valor Único: Este sistema transforma la firma digital de un costo fijo (SMS para todo) a un costo optimizado dinámico, reduciendo hasta 70% el costo operacional mientras mejora la experiencia del usuario mediante selección inteligente del mejor canal según contexto de la transacción.

Diferenciador Clave: Enrutamiento dinámico con reglas de negocio expresivas (SpEL), fallback automático resiliente, y modo degradado inteligente que previene cascadas de fallos en infraestructura crítica bancaria.

Project Classification

Technical Type: API Backend + Admin Web Portal

Domain: FinTech (Banking)

Complexity: High (Banking-grade compliance, multi-provider orchestration, event-driven)

Domain Context

Regulatory Environment:

- PCI-DSS:** Manejo de datos de transacciones financieras
- GDPR:** Protección de datos personales (pseudonimización obligatoria)

- **SOC 2 Type II:** Controles de seguridad y auditoría
- **Local Banking Regulations:** Compliance con regulaciones bancarias locales

Critical Constraints:

- **Zero Data Loss:** Garantía absoluta de entrega de eventos (transacciones no pueden perderse)
 - **Audit Trail:** Trazabilidad completa inmutable para auditorías regulatorias
 - **High Availability:** 99.9% uptime mínimo (downtime = imposibilidad de firmar transacciones)
 - **Security First:** Pseudonimización obligatoria, encryption at-rest, TLS everywhere
 - **Non-Repudiation:** Almacenamiento de pruebas criptográficas de providers
-

Success Criteria

Primary Success Metrics

1. Cost Optimization Achievement

- **Target:** Reducción de 50–70% en costos operacionales de firma digital
- **Measurement:** (Costo actual con SMS 100%) vs (Costo con routing inteligente)
- **Success:** Alcanzar ROI positivo en 3 meses

2. Operational Resilience

- **Target:** 99.9% de disponibilidad del servicio
- **Measurement:** Uptime mensual, tolerancia a fallos de providers
- **Success:** Cero incidentes críticos que bloqueen todas las transacciones

3. User Experience Preservation

- **Target:** < 300ms latencia P99 end-to-end
- **Measurement:** Tiempo desde solicitud hasta envío de challenge
- **Success:** Sin degradación perceptible vs solución anterior

4. Compliance Validation

- **Target:** Pasar auditorías PCI-DSS, GDPR, SOC 2
- **Measurement:** Audit findings, compliance gaps
- **Success:** Cero hallazgos críticos en auditorías

Business Impact Metrics

- **Adoption Rate:** 100% de transacciones críticas enrutadas por el sistema (no bypass)
 - **Provider Diversity:** Mínimo 3 providers activos simultáneamente
 - **Fallback Success Rate:** > 95% de firmas completadas incluso con fallos de providers
 - **Admin Efficiency:** Cambios de reglas sin código (tiempo de cambio < 5 minutos)
-

Product Scope

MVP - Minimum Viable Product

Core Capabilities (Release 1):

1. Signature Orchestration Engine

- Crear solicitudes de firma con contexto de transacción
- Evaluar reglas de routing (SpEL) para determinar canal óptimo
- Enviar challenges a través de provider seleccionado
- Gestionar ciclo de vida completo de firma (pending → sent → completed/failed/expired)
- TTL default 3 minutos por challenge

2. Multi-Provider Integration

- **SMS Provider:** Twilio (primary)
- **Push Provider:** In-app notifications (basic implementation)
- **Voice Provider:** Automated voice calls (future-ready interface)
- Abstracción hexagonal para agregar providers sin cambios en core

3. Routing Rules Engine

- Evaluación de expresiones SpEL contra contexto de transacción
- Prioridad de reglas (short-circuit: primera coincidencia gana)
- Reglas configurables vía Admin Portal
- Validación de sintaxis SpEL en tiempo real

4. Fallback & Resilience

- Cadena de fallback automático: PUSH → SMS → VOICE
- Retry con exponential backoff (max 3 intentos)
- Circuit breaker por provider (threshold 50% error rate)
- Degraded mode: pausar provider problemático por 5 minutos

5. Event-Driven Architecture

- Outbox pattern para garantía de eventos
- Debezium CDC: outbox_event → Kafka
- 8 eventos de dominio: REQUEST_CREATED, CHALLENGE_SENT, CHALLENGE_FAILED, PROVIDER_FAILED, SIGNATURE_COMPLETED, SIGNATURE_EXPIRED, SIGNATURE_ABORTED, ROUTING_RULE_CHANGED
- At-least-once delivery guarantee

6. REST API

- POST /api/v1/signatures - Crear solicitud (con idempotency-key)
- GET /api/v1/signatures/{id} - Consultar estado + routing timeline
- OpenAPI 3.1 specification completa
- OAuth2 JWT authentication

7. Admin Portal (React SPA)

- **Rule Management:** CRUD de routing rules con editor SpEL
- **Routing Timeline Viewer:** Visualización de intentos y fallbacks
- **Provider Health Dashboard:** Estado de providers, error rates, circuit breaker status
- **Roles:** Admin (full access), Auditor (read-only), Support (read signatures sin PII)

8. Observability Foundation

- Structured JSON logging (sin PII)
- Prometheus metrics: signature.created, challenge.sent, fallback.rate, provider.error_rate
- Distributed tracing (Jaeger/Zipkin)
- Health endpoint: /actuator/health

9. Security & Compliance

- Pseudonimización obligatoria de customer_id
- TLS/HTTPS everywhere
- HashiCorp Vault para credenciales de providers
- Audit log immutable con partition rotation
- Provider proof storage (non-repudiation)

Growth Features (Post-MVP)

Release 2 - Advanced Resilience:

- **Provider Health Monitoring:** Scheduled job cada 1 min para calcular error rates
- **Cost Optimization Dashboard:** Comparativa de costos por canal, savings vs all-SMS

- **A/B Testing Framework:** Experimentación con diferentes estrategias de routing
- **Advanced Fallback Rules:** Reglas personalizadas de fallback por contexto
- **Rate Limiting per Customer:** Protección contra abuso (10 signatures/min por customer)

Release 3 - Enhanced Channels:

- **Biometric Provider:** Integración con servicios de autenticación biométrica
- **Multi-Region Support:** Routing geográfico optimizado
- **Smart Routing ML:** Machine learning para optimizar selección de canal
- **Real-time Cost Tracking:** Dashboard con costos en tiempo real

Release 4 - Enterprise Features:

- **Multi-Tenant Support:** Aislamiento de reglas y configuración por tenant
- **Advanced Analytics:** Reporting avanzado, forecasting de costos
- **Compliance Automation:** Generación automática de reportes de compliance
- **Fraud Detection Integration:** Alertas basadas en patrones anómalos

Vision (Future)

- **Self-Learning Routing:** Sistema aprende de histórico para optimizar automáticamente
- **Predictive Maintenance:** Predicción de fallos de providers antes de que ocurran
- **Global Load Balancing:** Distribución inteligente de carga entre regions
- **Blockchain Audit Trail:** Immutable audit trail en blockchain para máxima garantía

Domain-Specific Requirements

Banking Compliance Requirements

PCI-DSS Compliance:

- Encriptación at-rest (TDE en PostgreSQL)
- Encriptación in-transit (TLS 1.3 mínimo)
- Audit logging de todos los cambios
- Separación de duties (roles admin vs auditor)
- Secret management via Vault (no secrets en código/config)

GDPR Compliance:

- Pseudonimización de customer_id (no PII en sistema)

- Right to be forgotten: Soft delete con retention policy
- Data minimization: Solo almacenar lo estrictamente necesario
- Audit trail de accesos a datos personales
- Data export capability (GDPR Article 20)

SOC 2 Type II Controls:

- Change management: Audit log de cambios en routing rules
- Access controls: RBAC con roles granulares
- Monitoring & alerting: SLO tracking ($P99 < 300ms$, availability $\geq 99.9\%$)
- Incident response: Degraded mode automático, alerting crítico
- Business continuity: Fallback chain, circuit breaker

Financial Services Specific

Non-Reputation:

- Almacenar `provider_proof` (recibo criptográfico del provider)
- Timestamp de todos los eventos con precisión de milisegundos
- Immutable audit log con hash chaining (opcional)

Transaction Integrity:

- Idempotency garantizada (Idempotency-Key en requests)
- Atomicidad: DB + Evento en misma transacción (Outbox pattern)
- Zero data loss: Kafka durability, min.insync.replicas=2

Operational Resilience:

- Graceful degradation (degraded mode cuando provider falla)
- No single point of failure (multi-provider architecture)
- Chaos engineering testing obligatorio antes de producción

API Backend Specific Requirements

API Architecture

RESTful Design:

- Resource-oriented URLs: `/signatures`, `/admin/rules`, `/admin/connectors`
- HTTP verbs semánticos: GET (idempotent), POST (create), PUT (update), DELETE

- HTTP status codes apropiados: 200/201 (success), 400 (bad request), 401 (unauthorized), 404 (not found), 503 (service unavailable)
- Hypermedia links: `Location` header en 201 Created

API Versioning:

- URL path versioning: `/api/v1/signatures`
- Contract-first development: OpenAPI 3.1 spec primero, código después
- Backward compatibility: Mantener v1 mientras clientes migren

Data Formats:

- Request/Response: JSON (Content-Type: application/json)
- Event payloads: Avro (Kafka Schema Registry)
- Date/Time: ISO 8601 format (RFC 3339)
- UUIDs: UUIDv7 (sortable, time-ordered)

Authentication & Authorization

Authentication Model:

- OAuth2 + JWT tokens (Bearer authentication)
- Token claims: `sub` (username), `roles` (array), `exp` (expiration)
- Token expiration: 1 hora (access token), 30 días (refresh token)
- Token refresh endpoint: `/api/v1/auth/refresh`

Authorization Model (RBAC):

- **Role: ADMIN** - Full access (manage rules, connectors, view all signatures)
- **Role: AUDITOR** - Read-only access (audit logs, metrics, no PII)
- **Role: SUPPORT** - Read signatures (tokenized customer_id, no sensitive details)
- **Role: USER** - Standard API access (create signatures, view own)

Security Headers:

- `Authorization: Bearer <jwt>`
- `X-API-Key: <api-key>` (alternative auth for service-to-service)
- `X-Trace-Id: <uuid>` (distributed tracing)
- `Idempotency-Key: <uuid>` (for POST requests)

Error Handling

Error Response Format:

```
{  
  "code": "VALIDATION_ERROR",  
  "message": "Invalid transaction context",  
  "details": [  
    {  
      "field": "transactionContext.amount.value",  
      "error": "Must be greater than 0"  
    }  
  ],  
  "timestamp": "2025-11-26T10:30:00Z",  
  "traceId": "550e8400-e29b-41d4-a716-446655440000"  
}
```

Error Codes:

- VALIDATION_ERROR – Input validation failed
- NOT_FOUND – Resource not found
- UNAUTHORIZED – Missing/invalid authentication
- FORBIDDEN – Insufficient permissions
- INTERNAL_ERROR – Unexpected server error
- SERVICE_UNAVAILABLE – All providers down (degraded mode)
- INVALID_SPEL – SpEL expression syntax error
- RATE_LIMIT_EXCEEDED – Too many requests

Rate Limiting

Global Rate Limits:

- Signature creation: 100 req/s per service (all customers combined)
- Admin operations: 10 req/s per admin user
- Query operations: 500 req/s per service

Per-Customer Rate Limits:

- Signature creation: 10 signatures/min per customer_id
- Prevent abuse and DoS attacks

Rate Limit Headers:

- X-RateLimit-Limit: 100
- X-RateLimit-Remaining: 95
- X-RateLimit-Reset: 1635789600 (Unix timestamp)

Rate Limit Exceeded Response:

- HTTP 429 Too Many Requests
- Retry-After: 60 (seconds to wait)

API Documentation

OpenAPI Specification:

- Complete OpenAPI 3.1 spec: `docs/architecture/05-api-contracts.yaml`
- Interactive docs: Swagger UI at `/swagger-ui.html`
- ReDoc alternative: `/redoc`

API Versioning Policy:

- Major version changes: Breaking changes (`/v2/`)
- Minor version changes: Backward-compatible additions (same `/v1/`)
- Deprecation notice: 6 months before removing old version

User Experience Principles

Admin Portal UX

Design Philosophy: "Banking-grade professionalism with modern SaaS simplicity"

Visual Identity:

- Clean, data-dense interface (admins are power users)
- Dark mode support (long sessions)
- Monospace fonts for code (SpEL expressions)
- Color coding: Green (healthy), Yellow (warning), Red (critical)

Key Interactions:

1. Rule Management Flow:

- List view: Sortable by priority, filterable by status
- Create/Edit: Inline SpEL validator with real-time syntax checking

- Test mode: "Dry run" con sample transaction context antes de guardar
- Bulk actions: Enable/disable multiple rules

2. Routing Timeline Visualization:

- Timeline vertical con eventos cronológicos
- Iconografía clara: ✓ (success), ✗ (failed), ⚠ (warning), ↺ (retry)
- Color coding por estado de challenge
- Expandible: Click para ver detalles de cada evento

3. Provider Health Dashboard:

- Card per provider con estado visual (traffic light)
- Real-time error rate graphs (Recharts)
- Circuit breaker state indicator
- Quick actions: Enable/disable provider, force health check

4. Navigation:

- Sidebar persistente con secciones principales
- Breadcrumbs para navegación profunda
- Search global: Buscar signatures por ID, customer, fecha

Mobile-First Considerations

Admin Portal es Desktop-First:

- Target: Admins en workstations
- Responsive: Tablets (iPad) supported, mobile phones degraded experience
- Rationale: SpEL editing, timeline visualization requieren espacio

Functional Requirements

FR Group 1: Signature Request Management

- **FR1:** El sistema puede recibir solicitudes de firma digital con contexto de transacción inmutable (JSONB)
- **FR2:** El sistema puede generar IDs únicos ordenables temporalmente (UUIDv7) para cada solicitud
- **FR3:** El sistema puede almacenar customer_id pseudonimizado (sin PII plain)
- **FR4:** El sistema puede generar hash SHA-256 del contexto de transacción para integridad
- **FR5:** El sistema puede establecer TTL default de 3 minutos por signature request
- **FR6:** El sistema puede consultar el estado actual de una signature request

- **FR7:** El sistema puede proporcionar timeline completo de routing y eventos
- **FR8:** El sistema puede abortar signature requests manualmente (admin intervention)
- **FR9:** El sistema puede expirar automáticamente signature requests al alcanzar TTL
- **FR10:** El sistema puede detectar y rechazar signature requests duplicadas (idempotency)

FR Group 2: Routing Decision Engine

- **FR11:** El sistema puede evaluar expresiones SpEL contra contexto de transacción
- **FR12:** El sistema puede aplicar reglas de routing en orden de prioridad (short-circuit)
- **FR13:** El sistema puede seleccionar canal óptimo basado en primera regla que coincide
- **FR14:** El sistema puede registrar qué regla determinó el routing (audit trail)
- **FR15:** El sistema puede manejar reglas sin coincidencias (default fallback to configured channel)
- **FR16:** El sistema puede validar sintaxis SpEL antes de persistir reglas
- **FR17:** El sistema puede deshabilitar/habilitar reglas sin eliminarlas
- **FR18:** El sistema puede reordenar prioridades de reglas
- **FR19:** El sistema puede almacenar metadata de quién creó/modificó cada regla

FR Group 3: Challenge Delivery

- **FR20:** El sistema puede enviar challenges SMS vía Twilio
- **FR21:** El sistema puede enviar push notifications vía Push Provider
- **FR22:** El sistema puede realizar llamadas de voz vía Voice Provider
- **FR23:** El sistema puede almacenar provider_challenge_id para correlación
- **FR24:** El sistema puede almacenar provider_proof (recibo criptográfico) para non-repudiation
- **FR25:** El sistema puede aplicar timeouts configurables por tipo de provider (5s external)
- **FR26:** El sistema puede registrar timestamp de envío y respuesta de challenges
- **FR27:** El sistema puede garantizar un solo challenge activo por signature request
- **FR28:** El sistema puede expirar challenges que superan TTL sin respuesta

FR Group 4: Fallback & Resilience

- **FR29:** El sistema puede detectar fallos de providers automáticamente
- **FR30:** El sistema puede intentar fallback a canal alternativo según cadena configurada
- **FR31:** El sistema puede crear nuevo challenge para cada intento de fallback
- **FR32:** El sistema puede aplicar retry con exponential backoff (max 3 intentos)

- **FR33:** El sistema puede calcular error rate por provider en ventana deslizante
- **FR34:** El sistema puede activar circuit breaker cuando error rate supera 50%
- **FR35:** El sistema puede pausar provider en degraded mode por duración configurable (5 min default)
- **FR36:** El sistema puede reactivar provider automáticamente tras periodo de pausa
- **FR37:** El sistema puede prevenir loops infinitos de fallback (max 3 canales)
- **FR38:** El sistema puede marcar signature request como FAILED si todos los fallbacks fallan

FR Group 5: Event Publishing

- **FR39:** El sistema puede persistir eventos de dominio en outbox table
- **FR40:** El sistema puede garantizar atomicidad entre cambio de estado y evento (misma TX)
- **FR41:** El sistema puede publicar eventos a Kafka vía Debezium CDC
- **FR42:** El sistema puede serializar eventos en formato Avro con schema validation
- **FR43:** El sistema puede particionar eventos por aggregate_id (ordering guarantee)
- **FR44:** El sistema puede incluir trace_id en eventos para distributed tracing
- **FR45:** El sistema puede publicar 8 tipos de eventos de dominio distintos
- **FR46:** El sistema puede almacenar hash de transaction context en eventos (sin PII)

FR Group 6: Admin - Rule Management

- **FR47:** Admins pueden crear routing rules con expresión SpEL
- **FR48:** Admins pueden validar sintaxis SpEL en tiempo real (pre-save)
- **FR49:** Admins pueden editar reglas existentes
- **FR50:** Admins pueden eliminar reglas
- **FR51:** Admins pueden habilitar/deshabilitar reglas individualmente
- **FR52:** Admins pueden asignar prioridades numéricas a reglas
- **FR53:** Admins pueden ver lista de todas las reglas ordenadas por prioridad
- **FR54:** Admins pueden filtrar reglas por estado (enabled/disabled)
- **FR55:** Admins pueden asociar descripción legible a cada regla
- **FR56:** Admins pueden especificar target channel por regla (SMS/PUSH/VOICE/BIOMETRIC)

FR Group 7: Admin - Provider Management

- **FR57:** Admins pueden ver lista de todos los providers configurados
- **FR58:** Admins pueden habilitar/deshabilitar providers manualmente
- **FR59:** Admins pueden ver error rate actual de cada provider
- **FR60:** Admins pueden ver estado de circuit breaker por provider
- **FR61:** Admins pueden ver última verificación de salud (health check timestamp)
- **FR62:** Admins pueden ver providers en degraded mode con timestamp de inicio
- **FR63:** Admins pueden forzar reactivación de provider en degraded mode
- **FR64:** Admins pueden ver configuración de cada provider (timeouts, retries)

FR Group 8: Admin - Monitoring & Visualization

- **FR65:** Admins pueden ver routing timeline de cualquier signature request
- **FR66:** Admins pueden buscar signature requests por ID
- **FR67:** Admins pueden ver métricas de cost optimization (savings vs all-SMS)
- **FR68:** Admins pueden ver distribución de challenges por canal (SMS/Push/Voice)
- **FR69:** Admins pueden ver tasa de fallback global
- **FR70:** Admins pueden ver dashboard de salud de providers en tiempo real
- **FR71:** Admins pueden exportar audit logs filtrados por fecha/entidad
- **FR72:** Admins pueden ver gráficos de error rate trending por provider

FR Group 9: Audit & Compliance

- **FR73:** El sistema puede registrar todos los cambios en routing rules en audit log
- **FR74:** El sistema puede registrar todos los cambios en connector config en audit log
- **FR75:** El sistema puede almacenar audit log en tabla separada con rotation mensual
- **FR76:** El sistema puede capturar IP address y user agent en audit log
- **FR77:** El sistema puede generar SHA-256 hash de transaction context (integrity check)
- **FR78:** El sistema puede almacenar provider proofs inmutables
- **FR79:** El sistema puede exportar audit log completo para auditorías externas
- **FR80:** El sistema puede retener audit logs según política de retention (365 días default)

FR Group 10: Security & Access Control

- **FR81:** El sistema puede autenticar usuarios vía OAuth2 JWT tokens
- **FR82:** El sistema puede validar roles en JWT claims (ADMIN, AUDITOR, SUPPORT)
- **FR83:** El sistema puede restringir endpoints /admin/* a role ADMIN

- **FR84:** El sistema puede restringir visualización de PII según role
 - **FR85:** El sistema puede aplicar rate limiting por customer_id (10/min)
 - **FR86:** El sistema puede aplicar rate limiting global (100/s creation)
 - **FR87:** El sistema puede almacenar secrets en Vault (no en DB/config)
 - **FR88:** El sistema puede rotar secrets de providers sin downtime
 - **FR89:** El sistema puede forzar TLS 1.3 mínimo en todas las conexiones
 - **FR90:** El sistema puede rechazar requests sin Idempotency-Key en POST
-

Non-Functional Requirements

Performance

Latency Requirements:

- **NFR-P1:** P99 latency end-to-end (request → challenge sent) < 300ms
- **NFR-P2:** P50 latency < 150ms
- **NFR-P3:** Database query timeout: 2 segundos máximo
- **NFR-P4:** Provider API call timeout: 5 segundos máximo
- **NFR-P5:** Kafka producer send timeout: 1.5 segundos máximo

Throughput Requirements:

- **NFR-P6:** Soportar 100 signatures/segundo sostenido
- **NFR-P7:** Soportar picos de 300 signatures/segundo durante 5 minutos
- **NFR-P8:** Connection pool: 20 conexiones máximo a PostgreSQL

Resource Limits:

- **NFR-P9:** Memory footprint < 2GB por instancia de aplicación
- **NFR-P10:** CPU utilization < 70% en operación normal

Security

Authentication & Authorization:

- **NFR-S1:** JWT tokens con RSA 256 signature mínimo
- **NFR-S2:** Tokens expiran en 1 hora (access) o 30 días (refresh)
- **NFR-S3:** RBAC enforcement en todos los endpoints
- **NFR-S4:** API keys rotables para service-to-service auth

Data Protection:

- **NFR-S5:** TDE (Transparent Data Encryption) habilitado en PostgreSQL
- **NFR-S6:** TLS 1.3 obligatorio para todas las conexiones externas
- **NFR-S7:** Secrets almacenados en HashiCorp Vault (no hardcoded)
- **NFR-S8:** Customer_id pseudonimizado (HMAC-SHA256 one-way)
- **NFR-S9:** Sin PII en logs, eventos, o métricas

Audit & Compliance:

- **NFR-S10:** Audit log immutable con timestamp preciso (millisecond)
- **NFR-S11:** Retention de audit logs: 365 días mínimo
- **NFR-S12:** Provider proofs almacenados indefinidamente
- **NFR-S13:** Compliance con PCI-DSS, GDPR, SOC 2 Type II

Security Testing:

- **NFR-S14:** Dependency scanning en CI/CD (OWASP Dependency Check)
- **NFR-S15:** Container scanning (Trivy) antes de deployment
- **NFR-S16:** Penetration testing semestral

Scalability

Horizontal Scaling:

- **NFR-SC1:** Backend stateless (escala horizontalmente sin sticky sessions)
- **NFR-SC2:** Soportar 3-5 réplicas de aplicación en Kubernetes
- **NFR-SC3:** Admin Portal servido desde CDN (estáticos)

Database Scaling:

- **NFR-SC4:** PostgreSQL con read replicas para queries (future)
- **NFR-SC5:** Connection pooling (HikariCP) con 20 max connections
- **NFR-SC6:** Index optimization para queries frecuentes

Event Streaming Scaling:

- **NFR-SC7:** Kafka con 12 partitions para signature.events topic
- **NFR-SC8:** Replication factor 3 para durability
- **NFR-SC9:** Min.insync.replicas = 2 para zero data loss

Availability & Resilience

Uptime Requirements:

- **NFR-A1:** 99.9% availability mensual (max 43 minutos downtime/mes)
- **NFR-A2:** Zero downtime deployments (rolling updates)
- **NFR-A3:** Graceful shutdown: 30 segundos para completar requests in-flight

Fault Tolerance:

- **NFR-A4:** Circuit breaker per provider (50% threshold → OPEN state)
- **NFR-A5:** Degraded mode automático (pausar provider 5 min tras circuit breaker)
- **NFR-A6:** Fallback chain completa (PUSH → SMS → VOICE)
- **NFR-A7:** Retry con exponential backoff (max 3 intentos)

Data Durability:

- **NFR-A8:** Zero data loss en eventos (Kafka durability + Outbox pattern)
- **NFR-A9:** PostgreSQL con daily backups (retención 30 días)
- **NFR-A10:** Point-in-time recovery capability (PITR)

Disaster Recovery:

- **NFR-A11:** RTO (Recovery Time Objective): < 1 hora
- **NFR-A12:** RPO (Recovery Point Objective): < 5 minutos
- **NFR-A13:** Multi-AZ deployment en AWS/GCP/Azure

Observability

Logging:

- **NFR-O1:** Structured JSON logging (Logback + Logstash encoder)
- **NFR-O2:** MDC context: traceId, signatureId, customerId (tokenized)
- **NFR-O3:** Log aggregation: ELK Stack o Loki
- **NFR-O4:** Log retention: 90 días

Metrics:

- **NFR-O5:** Prometheus metrics exportadas en /actuator/prometheus
- **NFR-O6:** Business metrics: signature.created, challenge.sent, fallback.rate
- **NFR-O7:** Technical metrics: provider.latency (P50/P95/P99), provider.error_rate
- **NFR-O8:** Grafana dashboards para SLO tracking

Distributed Tracing:

- **NFR-09:** Jaeger o Zipkin para distributed tracing
- **NFR-010:** Trace propagation a Kafka consumers
- **NFR-011:** Trace context en provider API calls

Alerting:

- **NFR-012:** Critical alerts: P99 > 300ms, availability < 99.9%, all providers down
- **NFR-013:** Warning alerts: Provider degraded mode, high fallback rate (>10%)
- **NFR-014:** Alert delivery: PagerDuty, Slack, Email

Integration

External Systems:

- **NFR-I1:** Twilio API integration con retry y circuit breaker
- **NFR-I2:** Push Provider API integration con timeout 3s
- **NFR-I3:** Voice Provider API integration con timeout 5s
- **NFR-I4:** HashiCorp Vault integration para secret management

Event Consumers:

- **NFR-I5:** Kafka events consumibles por múltiples servicios independientes
- **NFR-I6:** Avro schema evolution (backward compatibility)
- **NFR-I7:** Schema Registry para validación de eventos

API Integration:

- **NFR-I8:** OpenAPI 3.1 spec publicada y versionada
- **NFR-I9:** Swagger UI para testing interactivo
- **NFR-I10:** Client SDKs generables desde OpenAPI spec (future)

Este PRD captura la esencia de Signature Router & Management System – Una plataforma bancaria que optimiza costos de firma digital mediante routing inteligente multi-canal, garantizando resiliencia operacional y compliance regulatorio.

Created through BMAD Product Management workflow by AI Product Manager.